Cross Attention between Sound Event Localization and Detection
CS448 Final Project

Jeff Chang
Department of Electrical and Computer Engineering
University of Illinois at Urbana Champaign
Champaign, Illinois 61820
Email: kcchang3@illinois.edu

1. Introduction

The task of sound event localization and detection (SELD) includes two subtasks, estimation and sound event detection (SED) and direction of arrival (DOA). This project restricts the discussion of the SELD problem to perhaps the most popular formalization of this task: Task 3 of IEEE AASP Challenge on Detection and Classification of Acoustic Scenes and Events (DCASE2021). While different every year, datasets used for DCASE challenges essentially contain multiple sound event recordings of various categories, from multiple distances and directions of the array of microphones. These recordings are then emulated into sound scenes by filtering through room impulse responses, which varies in size, shape, acoustic reflective and absorptive properties. Figure 1 demonstrates the problem where data captured by the microphone may be subjected to interference from non-target sound classes or room acoustics. Proposed systems then has to solve the subtask of SED, which is to determine the occurrence and to predict the class of a sound event, and the subtask of DOA, which is to determine the Cartesian position of the sound event. Results are evaluated on their error rate (ER), F1-score (F), class-dependent localization error (LE), and localization recall (LR).

Classical implementation of a SELD system typically has one of the following structure.

1) A shared recurrent convolutional neural network (RCNN) follow by two fully connected (FC) branches for SED and DOA tasks.
2) Two separate branches of RCNN plus FC layers each targeting one of SED or DOA, with soft parameter sharing scheme such as cross stitch (2x2 matrix multiplication to weigh each branch's intermediate values).

With the wide use of transformers, I propose first replace recurrent layers with transformer layers, and then utilize cross attention within the transformer layers to replace the common cross stitch in structure 2 above. The project shows that this architecture can achieve comparable performance with state of the art solutions and out perform baseline by significant margins. Code for this project can be found at https://github.com/kaichieh121/SALSA

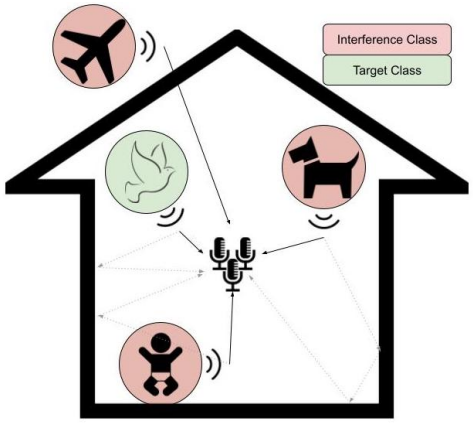

Figure 1. The SELD Problem

2. Background

The traditional SELDnet [1] (shown in figure 2) follows the first structure, where it relies on sharing a single convolutional recurrent neural network (CRNN) between the SED and DOA tasks for most of the feature extraction. The balance between multiple objectives can strongly affect the model performance. Shimada et al. [2] proposed ACCDOA, to only generate one 3 dimensional Cartesian target vector for DOA, where the length of the vector determines whether a sound class is active for the SED class. By simply changing this output format, Shimada showed the models are able to perform similarly to state-of-the-art systems in DCASE2020 while having less parameters.

Cao et al. proposed EINV2, a soft parameter sharing model. The model follows the second classic structure where it includes two branches (one for SED and one for DOA), each consists of CNNs and multi-head-self-attention (MHSA) blocks that share connections with the other branch by utilizing matrix multiplication to weigh values from each branch.

With above innovations of ACCDOA and EINV2, Shimada et al. [3] proposed 4 variants of D3Net architecture [4], a multidilated convolution DenseNet used for audio separation. These variants outputs either ACCDOA or track-

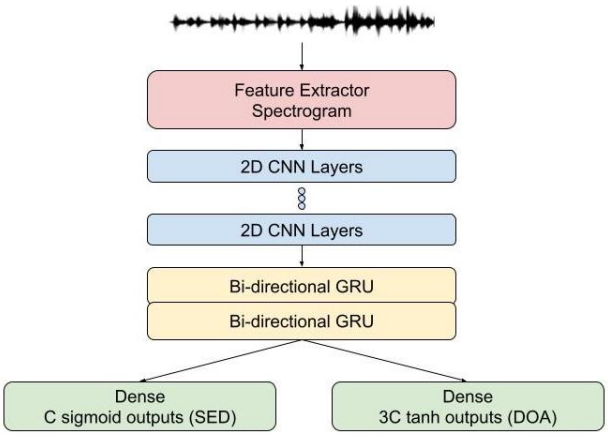

Figure 2. Basic SELDnet Architecture

wise outputs with different small architecture changes from D3Net. By ensembling these models, Shimada et al. were able to out perform all other solutions in DCASE2021, achieving state-of-the-art performance.

Nguyen et al. looked at the problem from the other side of the model, input features. The paper [5] proposed a novel feature named spatial cue-augmented log-spectrogram (SALSA), which essentially stacks the traditional spectrogram features, estimated direct-to-reverberant ratios, and principle eigenvectors of the spatial covariance matrix. With SALSA as input, [5] used a auto encoder structure to solve SELD. By modifying ResNet22 as encoder network, and used bidirectional recurrent layers and FC layers as decoder, this system was ranked second in DCASE2021.

3. Method

We propose a novel approach to SELD that is inspired by [5]. We used an encoder-decoder architecture where SALSA features were extracted from different input formats such as multichannel microphone array (MIC) or first-order ambisonics (FOA). Then two Resnet22-like architectures are used as encoder for each of the SED and DOA tasks. This is shown in Figure 3. We do not restrict our feature extractor to only extract SALSA, by experimenting with generalized cross-correlation phase transform (GCC-PHAT) or intensity vector (IV) on top of multichannel log-linear-spectrograms.

Mathematically, we can denote each input channel as $X_i \in \mathbb{R}$ where i represent either microphone number in MIC format or xyz axis in FOA format. Taking the Fourier Transform gives $\mathbf{X_i} = \mathcal{F}[X_i] \in \mathbb{C}^{T\times F}$ where T denotes the number of time frames and F denotes the number of features in each channel. We then compute each log linear spectrogram by

$$\hat{\mathbf{X}}_{\mathbf{i,spec}} = \log(|\mathcal{F}[\mathbf{X}]|^2) \in \mathbb{R}^{T\times F}$$

Similarly, we can compute the IV features from the 4 FOA inputs (omnidirectional, x, y, z), and the GCC-PHAT features for each pair (i, j) from the 4 microphones in the MIC inputs

$$\hat{\mathbf{X}}_{\mathbf{i,iv}} = -\frac{1}{\epsilon_0 c} Re\left[\mathbf{X}^*_{omni}\begin{pmatrix}\mathbf{X}_x\\ \mathbf{X}_y\\ \mathbf{X}_z\end{pmatrix}\right] \in \mathbb{R}^{T\times F}$$

$$\hat{\mathbf{X}}_{\mathbf{i,j,GCC-PHAT}} = \mathcal{F}^{-1}\left[\frac{\mathbf{X}_i\mathbf{X}_j^H}{||\mathbf{X}_i\mathbf{X}_j^H||}\right] \in \mathbb{R}^{T\times F}$$

SALSA features are calculated as described in [5]. By adding these additional features on top of the log-linear spectrograms, the input feature can be specified as $\hat{\mathbf{X}} \in \mathbb{R}^{B\times K\times T\times F}$. K denotes the number of channels (4 spectrogram channels and variable numbers of feature channels). B is batch size. By mean pooling after the last layer of the encoder, we have a hidden state $\mathbf{H} \in \mathbb{R}^{B\times T\times 512}$.

What set our work apart is the decoder architecture. We used two transformer networks in parallel, each targeting the SED or DOA task. Each transformer network consists of series of transformer blocks as described in classic Attention is All You Need [6]. Unlike other SELD solutions, we enable cross attention in the multiheaded attention layers of the SED and DOA branch. To the authors knowledge, there are currently no cross attention experimented in the context of SELD task. Let's define the query, key, and value matrix of each SED and DOA branch to be $\{Q, K, V\}_{SED}, \{Q, K, V\}_{DOA}$ respectively. We calculate the multihead attention layer output for SED as

$$\text{MultiHead} = \text{Concat}(\text{Attention}_1, ..., \text{Attention}_h)$$

The Attention layers can be self attention or cross attention depending on i, for odd i we have self attention and for even i we have cross attention

$$\text{SelfAttention}_i = softmax(\frac{Q_{SED}W_i^Q \times K_{SED}W_i^K}{\sqrt{d_k}})V_{SED}W_i^V$$

$$\text{CrossAttention}_i = softmax(\frac{Q_{SED}W_i^Q \times K_{DOA}W_i^K}{\sqrt{d_k}})V_{DOA}W_i^V$$

The multihead attention layer output for DOA branch will just mirror the equation above with SED hidden states replaced by DOA hidden states and vice versa.

We chose the most simple loss function: a weighted sum of SED prediction/target $\hat{\mathbf{Y}}_{SED}, \mathbf{Y}_{SED} \in \mathcal{R}^{B\times L\times C}$ and DOA prediction/target $\hat{\mathbf{Y}}_{DOA}, \mathbf{Y}_{DOA} \in \mathcal{R}^{B\times L\times 3C}$. L denotes the number of frames (different from T because of different input rate and label rate). C denotes the number of distinct classes.

$$\mathcal{L}(\hat{\mathbf{Y}}, \mathbf{Y}) = \alpha\mathcal{L}_{BCE}(\hat{\mathbf{Y}}_{SED}, \mathbf{Y}_{SED}) + \beta\mathcal{L}_{MSE}(\hat{\mathbf{Y}}_{DOA}, \mathbf{Y}_{DOA})$$

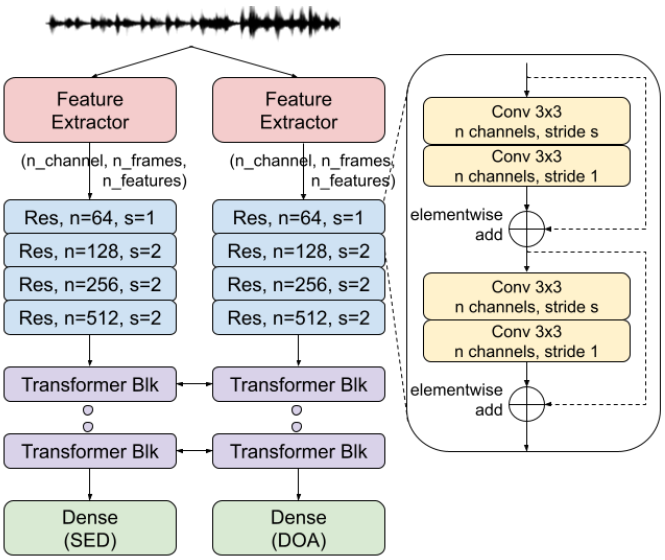

Figure 3. Proposed Architecture

4. Experiment

4.1. Dataset

We adopted the dataset used in DCASE2021 since many result from DCASE2022 has not been published and a full scale comparison may be difficult. DCASE2021 also has much more data than DCASE2022 because DCASE2022 relies on measuring actual rooms when DCASE2021 simulates the data with in room responses. The dataset contains audio recordings that has multiple sound events from 12 classes coming at different direction at different times. Room information is simulated by convolving the audio with in room responses. The dataset comes in 2 format: multichannel microphone array (MIC) or first-order ambisonics (FOA). As described in section 3, we extract GCC-PHAT features from MIC inputs and IV from FOA inputs.

4.2. Metric

We follow the exact 4 official metrics provided by DCASE2021 challenge. Let's first define typical statistics in the SELD context. At each time frame, we can calculate the true positives (TP), false positives (FP), and false negatives (FN) for each class. An angular threshold of 20° is applied to predictions in comparison to the reference sound events. From these, we can calculate the following.

$$\text{Location-dependent F1-score } F_{\leq 20^\circ} = \frac{2 * TP_c}{2 * TP_c + FP_c + FN_c}$$

$$\text{Error rate } ER_{\leq 20^\circ} = \frac{\sum_{k=1}^{K} E(k)}{\sum_{k=1}^{K} N(k)}$$

$$\text{Localization Error } LE_c = \sum_{i=1} \theta_i / TP_c$$

$$\text{Localization recall } LR_c = \frac{TP_c}{TP_c + FN_c}$$

where $E(k)$ is the number of wrongly predicted events and missed events and $N(k)$ is the total number of reference activities at a frame for class k. θ_i refers to the angular error for the ith prediction and target.

4.3. Code Setting

The proposed model was trained against 2 baselines under the exact same setting, with the same epochs and hyperparameters. The first baseline was the ResNet22+GRU architecture proposed by Nguyen et al [5]. We ran their publish code on the dataset and report the results here. We implemented the SELDnet baseline [1] under the same code base for easier comparison, since the data augmentation done in SALSA showed to improve the SELDnet performance by a little. Finally, our proposed architecture was also ran under the same training code. We ran experiments for both MIC and FOA inputs formats as well as both loss functions described in section 3. Furthermore, we experimented with different numbers of transformer layers and cross attention layers as ablation studies.

5. Discussion

In terms of the goal of this paper, which it to prove the potential of using transformer layers with cross attention between SED and DOA tasks, and to create an architecture that can outperform the SELD baseline, we claim the results to be successful.

Table 1 gave us a good idea on how well the proposed architecture work as compared with the state of the art Resne-based architecture proposed by Nguyen et al [5], as well as the baseline SELDNet [1]. We see that our model out performs SELDNet in every single metric with considerable margin. Specifically, it increase the performance in error rate by 30%, F1 by 42%, localization error by 51%, and localization recall by 9.8%. This alone is more than enough to demonstrate the usefulness of using ResNet alongside with transformers and cross attention to share weights between the SED task branch and DOA task branch. This is not to mention the performance of the proposed architecture gets fairly comparable to SOTA model, with only difference within the thousands decimal place, which is well within the error margin between different random weights initialization. However, it is not quite clear if cross attention itself has any benefits as it does not out perform SOTA with almost double the number of parameters.

Table 2 gave us further insight on the effectiveness of cross attention transformer layers. When comparing the SELD performance between the same architecture with and without cross attention for both 4 layers and 12 layers, we see that cross attention bring benefits to almost every metric. Beyond demonstrating the effectiveness of this decoder architecture to connect SED and DOA tasks, this shows that cross attention transformer layers can potentially correlate

TABLE 1. RESULT FOR DIFFERENT INPUT FORMATS

Model	Input	$\downarrow ER_{\leq 20^\circ}$	$\uparrow F_{\leq 20^\circ}$	$\downarrow LE$	$\uparrow LR$
Nguyen et al	FOA	**0.423**	**0.690**	**14.6**	**0.717**
SELDnet	FOA	0.709	0.458	24.7	0.613
Proposed	FOA	0.497	0.649	12.1	0.680
Nguyen et al	MIC	**0.429**	**0.695**	**12.8**	**0.719**
SELDnet	MIC	1.00	0.037	66.8	0.137
Proposed	MIC	0.602	0.586	17.7	0.712

TABLE 2. RESULT FOR VARIATION OF PROPOSED METHOD

N	cross	$\downarrow ER_{\leq 20^\circ}$	$\uparrow F_{\leq 20^\circ}$	$\downarrow LE$	$\uparrow LR$
2	no cross	0.504	0.625	17.0	0.693
4	no cross	0.534	0.617	16.7	0.692
12	no cross	0.534	0.598	15.6	0.650
4	0,2	0.514	0.631	14.8	0.691
4	1,2	0.552	0.624	17.0	**0.739**
4	1,3	0.537	0.610	17.1	0.702
12	1,5,9	**0.497**	**0.649**	**12.1**	0.680
12	4,5,6,7	0.517	0.632	12.2	0.659

other two different but related tasks, and eventually help the performance of each.

While it may seems like our model has the potential to outperform SOTA, we would like to reiterate that the encoder structure of our solution is a double duplication of that in SOTA. This means the slight performance degradation could be caused by our innovative decoder not suitable for the SELD task. The idea of transformer layers arranged in this way was inspired by wav2vec 2.0, which was an important innovation in the field of automatic speech recognition (ASR). However, those layers are typically pretrained on thousands of hours of data prior finetuning on the specific tasks. No pretraining was done in our experiments, so it might be difficult to learn more important attention correspondence to improve upon the SOTA. We defer this task to future work because we were not trying to outperform SOTA in this project, but to show the effectiveness of weight sharing by cross attention between SED and DOA tasks. To the best of our knowledge, pretraining a transformer network almost always bring benefits to downstream tasks, so we believe further development of our proposed architecture has the potential to beat SOTA.

6. Conclusion

In this paper, we proposed a model architecture that utilize the same encoder as the SOTA solution to the SELD tasks, and a cross attention transformer layers as its decoder. We were able to show the cross attention mechanism can share information between SED and DOA tasks, and ultimately bring benefits to the overall performance. By significantly outperforming the baseline, and achieved comparable results with SOTA, we conclude this architecture choice to be successful. We notice the potential for this kind of architecture to perform even better if we were to introduce another stage of pretraining such as those done in wav2vec 2.0. We defer this to future work but we are very optimistic with that setting to reach and even surpass SOTA.

References

[1] S. Adavanne, A. Politis, J. Nikunen, and T. Virtanen, "Sound event localization and detection of overlapping sources using convolutional recurrent neural networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 1, pp. 34–48, mar 2019.

[2] K. Shimada, N. Takahashi, Y. Koyama, S. Takahashi, E. Tsunoo, M. Takahashi, and Y. Mitsufuji, "Ensemble of accdoa- and einv2-based systems with d3nets and impulse response simulation for sound event localization and detection," 2021.

[3] ——, "Ensemble of accdoa- and einv2-based systems with d3nets and impulse response simulation for sound event localization and detection," 2021.

[4] N. Takahashi and Y. Mitsufuji, "Densely connected multidilated convolutional networks for dense prediction tasks," 2020.

[5] T. N. T. Nguyen, K. Watcharasupat, N. K. Nguyen, D. L. Jones, and W. S. Gan, "Dcase 2021 task 3: Spectrotemporally-aligned features for polyphonic sound event localization and detection," 2021.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017.